

In this laboratory, you will build an arena to run a different behavioural-based controller. Previously, our robot could navigate through a line circuit by using only its ground sensors. The robot was also capable of detecting but not avoiding obstacles using distance sensors. Now, we want the robot to navigate in the direction of a source of light (e.g.: phototaxis).

NOTE: For **Edinburgh Students ONLY**: If you need help during a timetabled lab session, please fill in [THIS FORM](#) and someone will get back to you as soon as possible. Please note that this form **should be used** only during the timetabled lab session (requests outside the timetabled lab session will not be attended to).

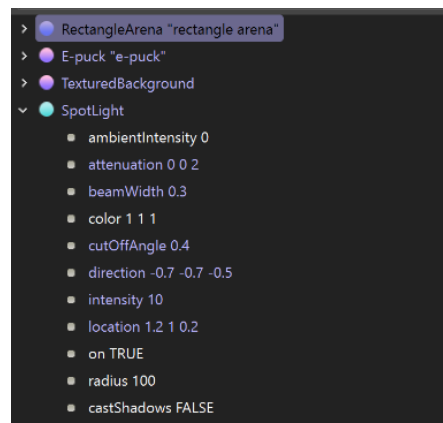
Alternatively, if you are at the GRID Lab at the HWU campus, please raise your hand from your desktop station and somebody will come over to help you.

Activity 1: Creating the arena and adding the e-puck robot

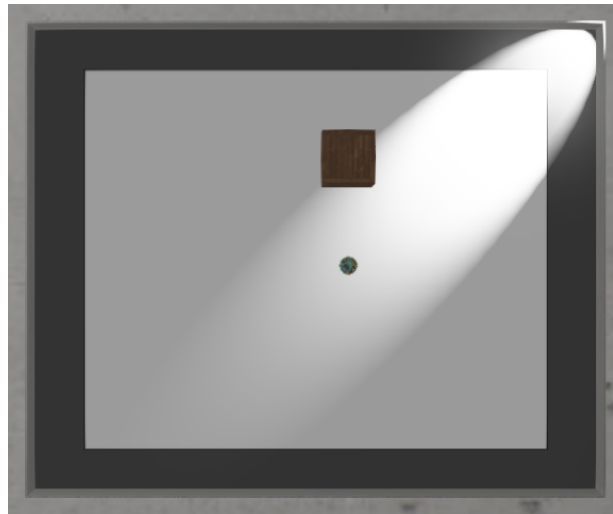
First, you will have to create a new project with a new world file name (e.g., lab2.wbt). Then, you need to create the arena and add the e-puck robot. To do that, please follow the steps from our previous Lab 1 (i.e, Lab 1: Activities 1 to 3). But this time, instead of adding the line circuit “circuit_line_lab1.jpg”, and the robot’s controller from our Lab 1, use the file “floor_texture_lab2.png” and the robot controller code “e-puck_light_lab2.py”, all available on CANVAS (Lab 2).

Activity 2: Adding a different source of light to the arena

Another important change is the type of light. You will add a ‘SpotLight’ node instead of the ‘PointLight’ node. Please, set up the light node using the same values from the figure below.



Your new arena now should look like the figure below.



TIP: You can play with the value of the fields “location” and “direction” and check what happens with the spotlight in the arena.

Add the controller code “e-puck_light_lab2.py” that you can find on CANVAS (Lab 2) to the e-puck node. Once the new controller is activated, the robot will always move towards the source of light.

NOTE: If the robot light sensors do NOT detect any source of light it will NOT move.


TIP: The light sensors can detect the light source only when (sometimes) it is very close to the source of the light beam. You can play with the initial position of the robot and observe when the robot detects the light and starts moving towards it.

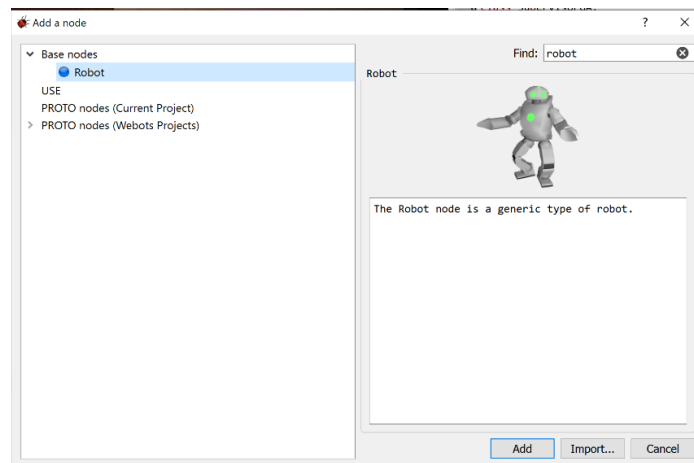
Activity 3: Adding a Supervisor node

Now suppose you want the light source to appear at a different “location” in the arena automatically. In order to do that we need to create a Supervisor node to move the position of the spotlight. For example at every minute (60s) of the experiment.

Remember that to move the spotlight inside the Supervisor code, you need to define the “location” and “direction” fields. Which are vectors of 3 values indicating the x,y,z positions.

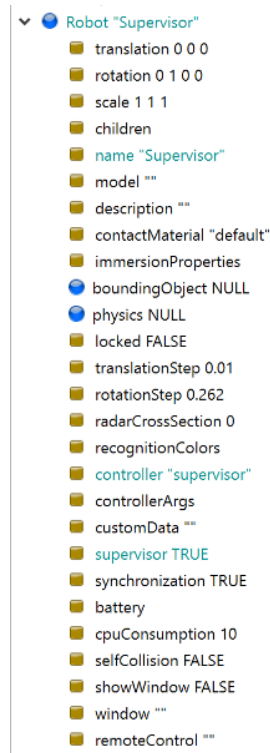
To create the Supervisor node:

Click on any node in the [ST] (e.g. ‘Worldinfo’) and click on the add button () located at the top bar of Webots. Search for the ‘Robot’ node and add it to your project. For further information on Supervisor nodes, please check the following link (<https://cyberbotics.com/doc/reference/supervisor>).

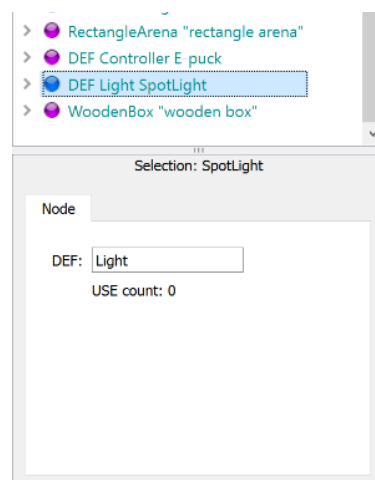


Once it has been created, you need to change a few fields of this node (see figure below). Please, change the name of the node to ‘Supervisor’, set the field ‘supervisor’ to ‘True’ and select the controller file “supervisor_lab2.py” from our CANVAS.

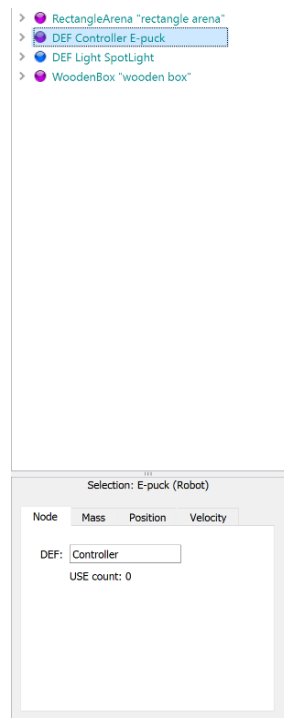
Follow the steps from Activity 6 of Lab 1 if you do not remember how to load a controller into your node.



Finally, you need to give a name to the 'SpotLight' node. Double-click on it and put on the DEF box the name 'Light' (see figure below).



And, you need to do the same thing for the robot's node. Set the name 'Controller' on it (see figure below).



Those names are important because you will need to identify both nodes through your Supervisor controller's code. In order to find them, you look by their names. Only then, you will be able to reset the robot's position and the "location" and "direction" of the source of light on the Supervisor's code, where it is written:

```
#####

## Add your code here ##

#####
```